

## **BAB 3**

### **METODOLOGI PENELITIAN**

#### **3.1 Metodologi Penelitian**

Pada bagian ini dijabarkan langkah-langkah yang hendak dilakukan dalam menyusun dan mengerjakan penelitian. Langkah-langkahnya sebagai berikut:

##### **1. Studi Literatur**

Studi literatur dilakukan untuk mempelajari dan memperdalam teori yang terkait dengan metode yang dapat diimplementasikan dalam rancang bangun push notification pada aplikasi toko grosir berbasis Android. Tahap ini dilakukan dengan melakukan studi dari berbagai referensi seperti jurnal, paper dan referensi lain dari penelitian terkait.

##### **2. Analisis Kebutuhan**

Dalam analisis kebutuhan hal yang perlu dilakukan untuk mengetahui apa yang dibutuhkan dalam melakukan implementasi algoritme ke dalam sistem, penentuan perangkat lunak dan perangkat keras juga diperhitungkan untuk kebutuhan penggunaan kebutuhan pada aplikasi.

##### **3. Perancangan dan Pembuatan Aplikasi**

Perancangan pembuatan aplikasi dilakukan agar rancangan aplikasi dapat dibuat sesuai dengan kebutuhan dan memiliki proses terstruktur. Tahapan ini mencakup pembuatan flowchart, lalu diikuti dengan pembuatan aplikasi berdasarkan hasil perancangan.

4. Pengujian dan Hasil Evaluasi Dalam tahap pengujian, pengujian dilakukan untuk memastikan aplikasi berjalan dengan baik dan memiliki fungsi yang sesuai dengan tujuan awal. Evaluasi hasil pengujian juga dilakukan untuk menguji hasil aplikasi.

### **3.2 Analisis Kebutuhan Sistem**

Analisis kebutuhan pada sistem, dapat menentukan seluruh kebutuhan yang ada pada sistem secara lengkap. Analisis kebutuhan sistem dibagi menjadi dua yaitu analisis kebutuhan non fungsional dan analisis kebutuhan fungsional.

#### **3.2.1 Analisis Kebutuhan Non Fungsional**

##### *1. Operational*

##### *(a) Perangkat Keras dan Lunak untuk membangun sistem*

- Sistem dapat berjalan pada operasi Windows 7/8/10 (32 atau 64 bit).
- Tools untuk membangun aplikasi android adalah Android Studio 4.2.1.
- Menggunakan XAMPP v3.2.2 sebagai *web server*.
- Visual Studio Code sebagai teks editor
- Menggunakan RAM atau memori minimal 3 GB. Direkomendasikan RAM 8 GB.
- *Minumium* memiliki 2 GB penyimpanan pada hardisk yang tersedia. Direkomendasikan 4 GB (500 MB untuk penyimpanan IDE dan 1,5 GB. untuk penyimpanan Android SDK dan sistem emulator).
- Resolusi layar *minimum* 1280 x 800.

- Spesifikasi *processor* intel *core i3 minimum* dan *core i5 recommended*.

(b) Perangkat Keras dan Lunak untuk penggunaan sistem.

- Berjalan pada Smartphone Galaxy A3 *minimum* dan sejenisnya.
- Sistem Operasi Android versi 5.0 Lollipop.

## 2. *Security*

(a) Terdapat *username* dan *password* untuk membuka sistem.

## 3. *Information*

(a) Ditampilkan informasi apabila *input* data pada saat melakukan *register* salah, atau sudah pernah digunakan.

(b) Ditampilkan informasi apabila *input username* dan *password* salah.

(c) Ditampilkan informasi apabila pembeli memasukan barang ke *cart*.

(d) Ditampilkan informasi apabila pembeli melakukan *checkout* barang.

## 4. *Performance*

(a) Aplikasi dapat digunakan secara *online* dengan data produk yang tersinkron dengan *web server*.

### 3.2.2 Analisis Kebutuhan Fungsional

1. Sistem yang digunakan harus memiliki Access Control List (*username* dan *password*).

2. Sistem dapat melakukan *input* data.

(a) Admin dapat melakukan *input* kategori baru.

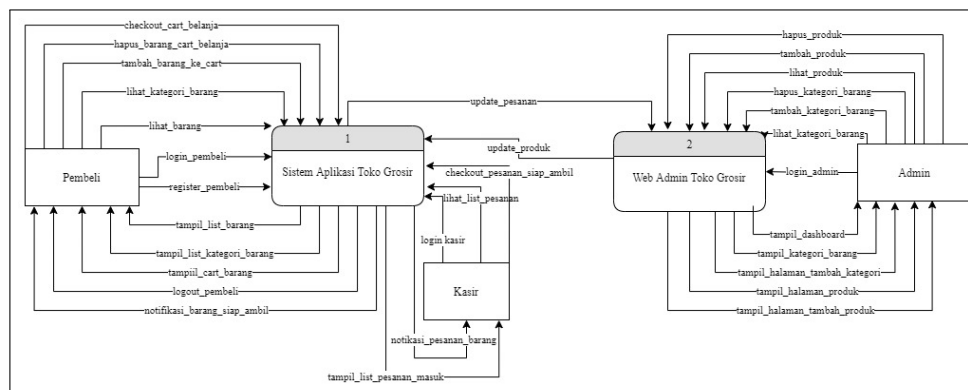
(b) Admin dapat melakukan *input* produk baru.

3. Sistem dapat melakukan *push notification* dari pembeli ke kasir untuk orderan baru.
4. Sistem dapat melakukan *push notification* dari kasir ke pembeli untuk informasi pengambilan barang.

### 3.3 Perancangan Aplikasi

Perancangan sistem aplikasi yang dibuat akan direpresentasikan menggunakan *Data Flow Diagram* dan flowchart. Terdapat beberapa flowchart yang akan menjelaskan alur kerja aplikasi secara keseluruhan, kemudian untuk setiap modul flowchart akan dijelaskan kembali secara detail pada tahap selanjutnya. Dalam perancangan aplikasi terdapat pembuatan tampilan aplikasi atau disebut juga *User Interface*.

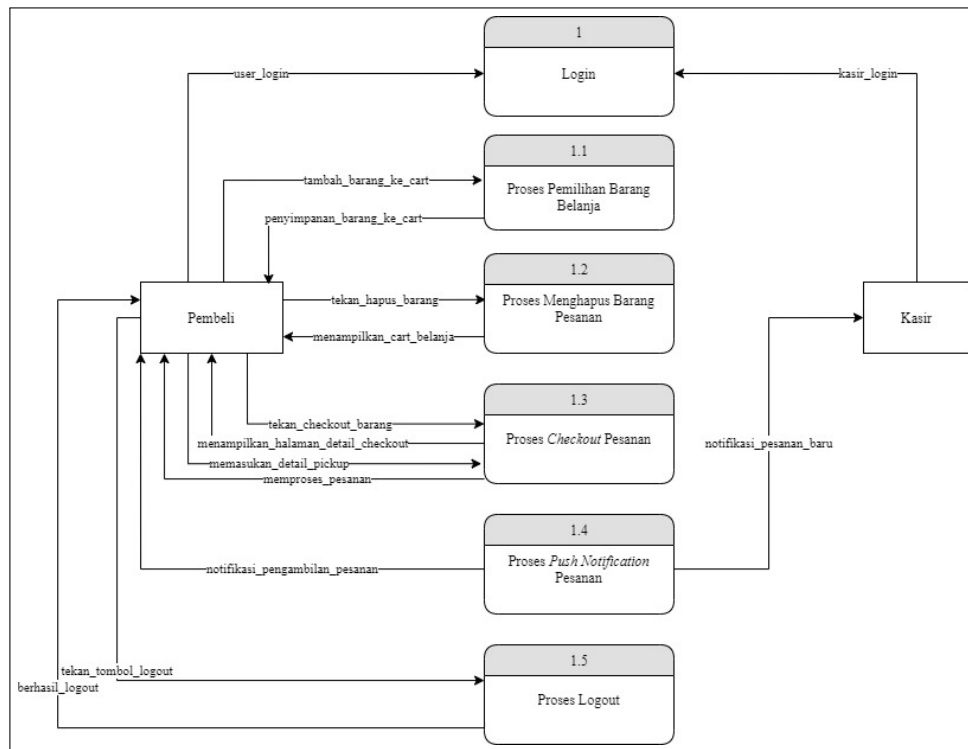
#### 3.3.1 Data Flow Diagram



Gambar 3.1. *Data Flow Diagram* Level 0

Pada gambar 3.1 dapat dilihat *Data Flow Diagram* level 0. Diagram ini menjelaskan secara garis besar alur dari jalannya aplikasi yang dibuat. Terdapat beberapa entitas yaitu User, Kasir, dan juga Admin. Entitas *User* ini bisa melakukan login atau *register*, memilih untuk melihat barang, detail barang, memasukkan barang ke *cart* belanja, dan menghapus *cart* belanja serta melakukan *checkout* belanja dan

menerima notifikasi dari sistem bahwa barang sudah siap diambil. Sedangkan kasir juga bisa melakukan login, melihat list pesanan dari *user*, melakukan *checkout* untuk pesanan siap diambil dan menerima notifikasi bahwa ada orderan baru masuk. Dan untuk entitas yang terakhir yaitu admin, admin dapat melakukan *input* kategori, *input* barang sesuai kategori yang ada, dan melakukan update stok dan harga.



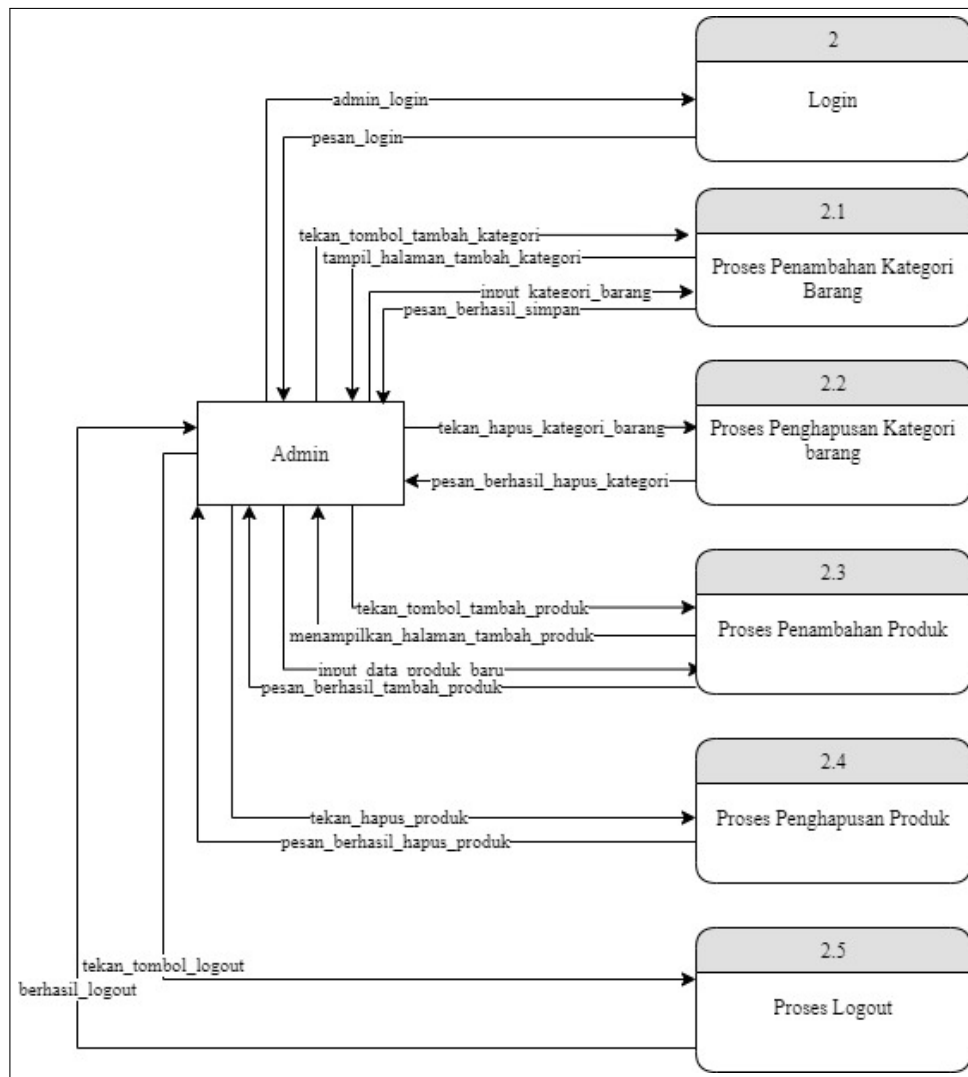
Gambar 3.2. *Data Flow Diagram Level 1*

Kemudian dilanjutkan dengan *Data Flow Diagram Level 1* di mana berisikan flow pada aplikasi yang terjadi antar pembeli dan penjual. Terdapat lima proses yang terjadi di dalam sistem aplikasi seperti pada gambar 3.2. Berikut 5 proses yang ada di dalam aplikasi tersebut.

1. Proses *Login* di mana *user* atau kasir diharuskan untuk memasukkan data *username* atau *password* yang sudah terdaftar untuk *login*.
2. Proses pemilihan barang belanja yaitu proses di mana pembeli memilih barang apa saja yang ingin dibeli, dengan masuk ke detail barang dan memasukkan jumlah barang yang ingin dibeli, ketika sudah memasukkan jumlah

yang ingin dibeli pembeli menekan tombol simpan, dan data pesanan barang akan di tampung di dalam *cart* belanja

3. Proses menghapus barang pesanan yaitu proses untuk menghapus barang yang tidak jadi dibeli dari list *cart*
4. Proses *checkout* pesanan, proses ini digunakan untuk saat pembeli sudah yakin dengan apa yang akan dibeli, pembeli melakukan *checkout* pesanan di mana pesanan tersebut akan dikirimkan ke kasir di toko untuk disiapkan barang pesanan tersebut
5. Proses *Push Notification* Pesanan yaitu proses pengiriman notifikasi antar user. Di mana penjual / kasir akan menerima notifikasi yang berisikan pesanan pembeli untuk disiapkan. Lalu jika sudah disiapkan penjual / kasir dapat mengirimkan notifikasi untuk pembeli bahwa barang telah disiapkan dan dapat diambil.
6. Proses Logout, proses terakhir di mana terjadi apabila pembeli atau kasir menekan tombol Logout, setelah tombol tersebut ditekan akan kembali ke halaman login.



Gambar 3.3. Data Flow Diagram Level 2

Pada gambar 3.3 Data Flow Diagram level 2 terdapat juga lima proses yang terjadi di bagian admin berikut proses yang terjadi.

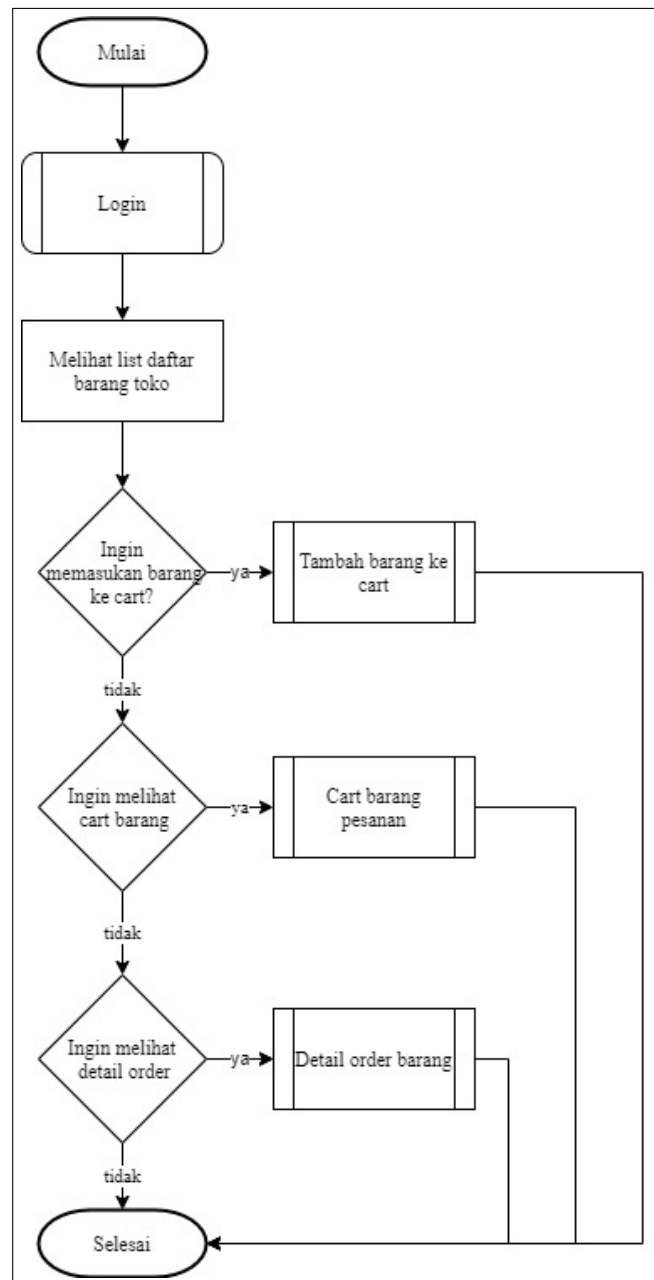
1. Proses *Login* yang merupakan proses di mana admin perlu memasukkan *user-name* dan *password* untuk *login* dan masuk ke halaman dashboard.
2. Proses penambahan kategori barang yaitu proses di mana admin melakukan *input* berupa nama kategori dan keterangan untuk kategori tersebut
3. Proses penghapusan Kategori barang, dalam proses ini admin dapat menghapus kategori yang sudah tidak ada atau tidak dijual lagi di toko tersebut.

4. Proses penambahan produk, dalam proses ini admin melakukan penambahan produk yang akan dijual. Di mana admin membutuhkan *input* kategori barang yang diambil dari kategori yang sudah terdaftar, nama produk, deskripsi - produk, harga, jumlah stock, dan foto produk. Setelah memasukkan data yang diperlukan, akan disimpan ke dalam database.
5. Proses penghapusan produk di mana proses untuk menghapus produk yang sudah tidak dijual lagi oleh toko.
6. Proses Logout yaitu proses terakhir di mana terjadi apabila admin menekan tombol logout, dan akan kembali ke halaman login.

### 3.3.2 Flowchart Aplikasi

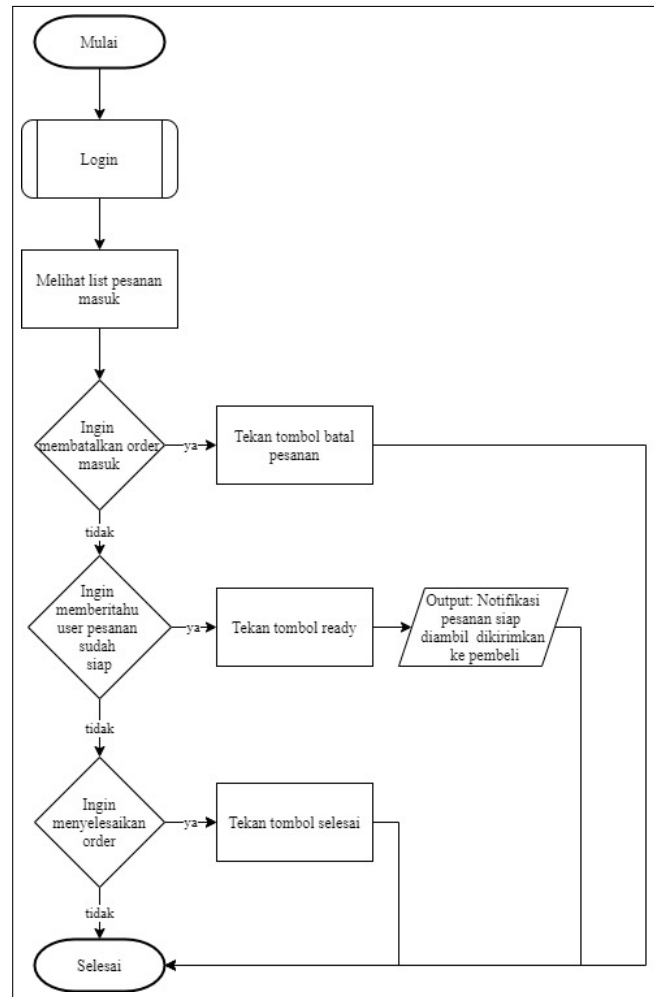
Terdapat tiga proses awal yaitu sebagai pembeli, kedua sebagai kasir, dan ketiga sebagai admin. Proses awal yang dijalankan di aplikasi sebagai pembeli adalah proses *login*. Di mana *user* diminta untuk melakukan *input* berupa *username* dan *password* yang telah terdaftar. Setelah berhasil melewati proses *login* maka akan menuju halaman utama. Di halaman utama terdapat lima menu yaitu menu *home*, menu produk, menu order, menu *account*, dan menu *about*. Proses kemudian masuk ke fungsi dari aplikasi sebagai pembeli yang terdiri dari beberapa modul yaitu modul tambah barang ke *cart*, modul *cart* barang pesanan, dan modul detail order barang. Proses ini dapat dilihat di Gambar 3.4.





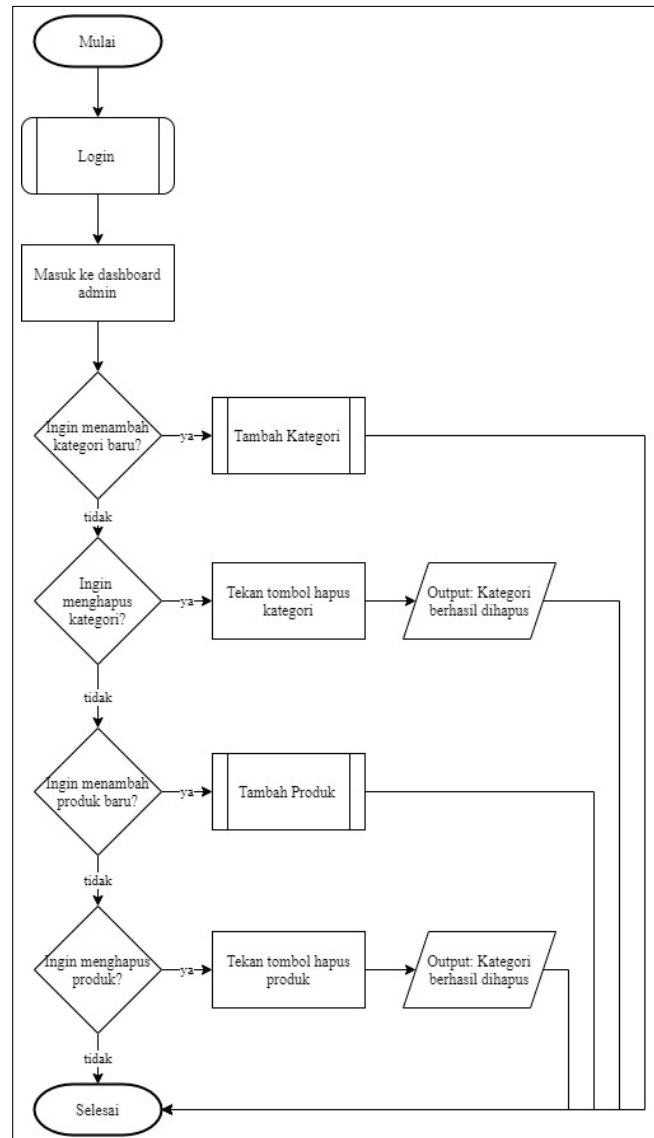
Gambar 3.4. Flowchart Awal Aplikasi sebagai pembeli

Proses yang kedua adalah sebagai kasir, di mana proses awalnya sama seperti pembeli yaitu melakukan *login*. Di mana *user* diminta untuk memasukkan *username* dan *password* yang telah terdaftar. Setelah berhasil melewati proses *login* maka akan menuju halaman utama. Di halaman utama kasir terdapat menu order, menu *history*, menu *account* dan menu *about*. Proses kemudian masuk ke fungsi dari aplikasi sebagai kasir. Proses ini dapat dilihat di Gambar 3.5



Gambar 3.5. Flowchart Awal Aplikasi sebagai kasir

Proses yang ketiga adalah sebagai admin, di mana proses awalnya seperti pembeli dan kasir yaitu melakukan *login*. Di mana *user* diharuskan untuk memasukkan *username* dan *password* yang telah terdaftar. Setelah berhasil melakukan proses *login* maka akan menuju halaman utama. Di halaman utama admin terdapat menu kategori yang dapat menambah kategori baru, dan menu produk yang terdapat fungsi tambah produk. Proses kemudian masuk ke fungsi dari aplikasi sebagai admin yang terdiri dari beberapa modul yaitu tambah kategori dan modul tambah produk. Proses ini dapat dilihat di Gambar 3.6

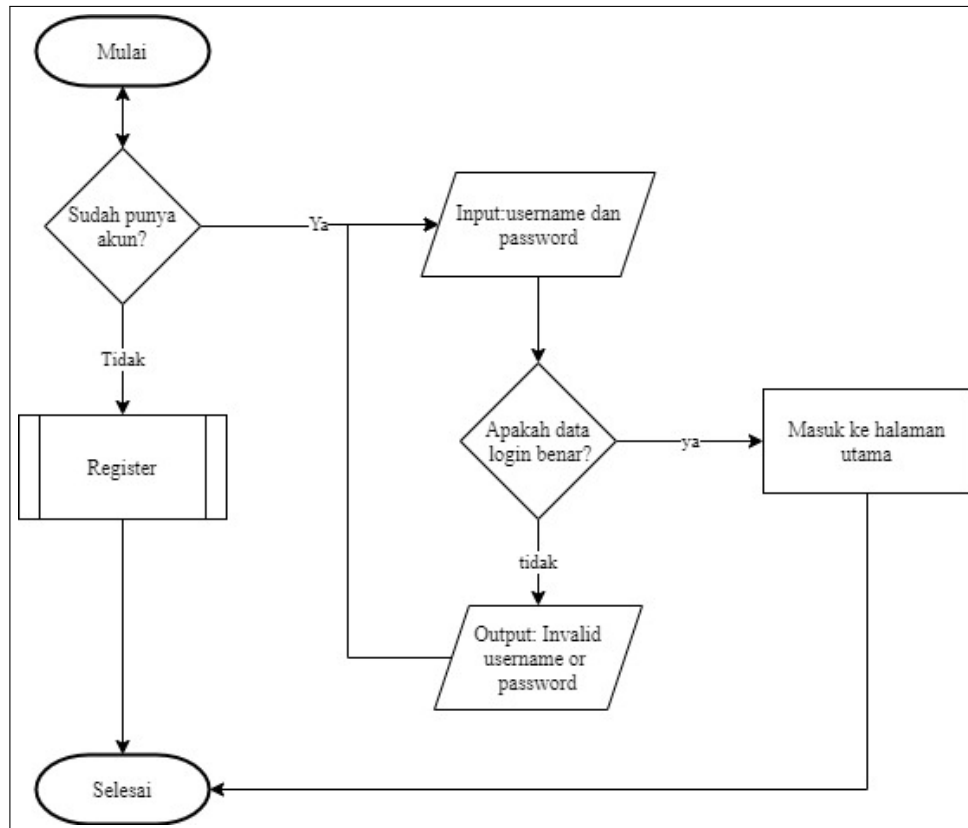


Gambar 3.6. Flowchart Awal Aplikasi sebagai admin

Selanjutnya setiap modul yang ada pada flowchart awal aplikasi akan dijelaskan secara lebih lengkap sebagai berikut.

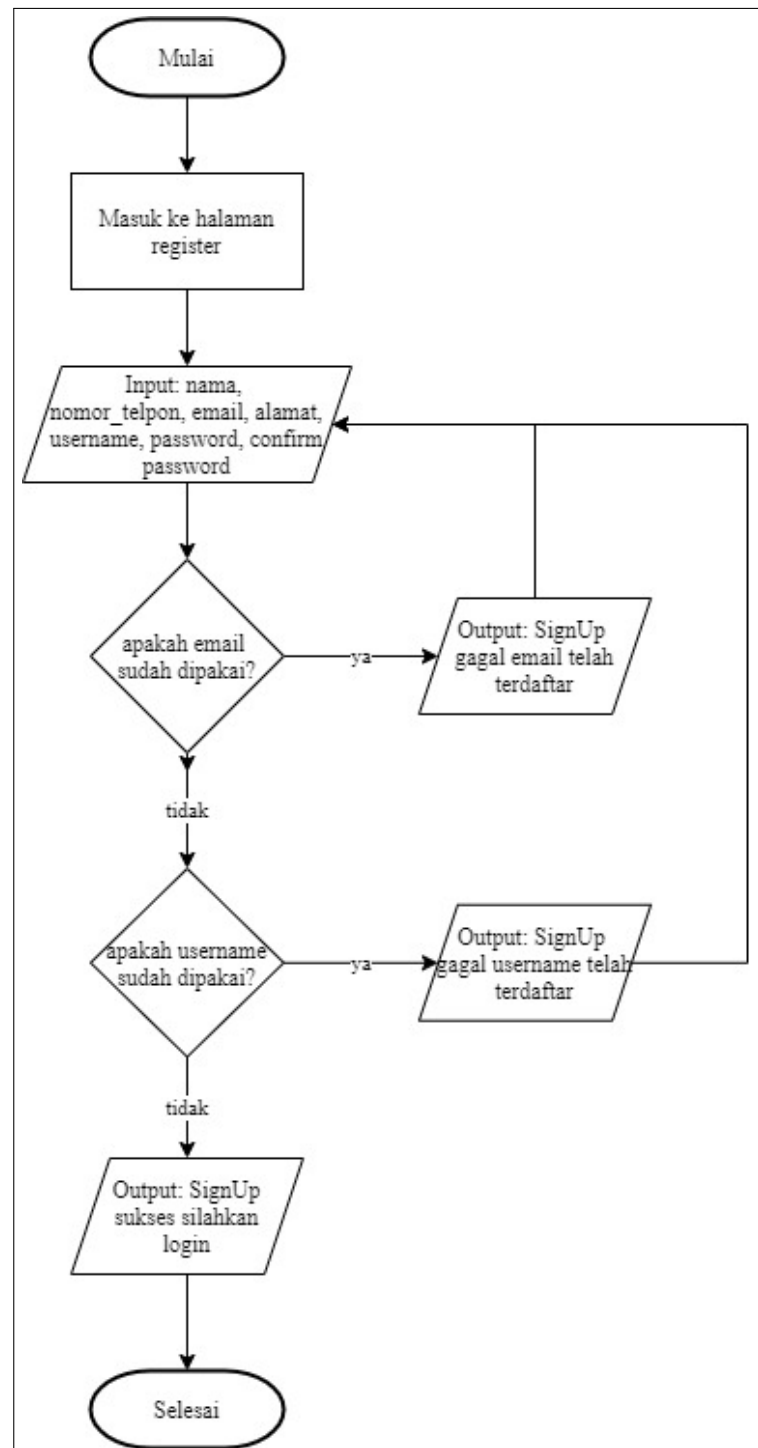
### A Flowchart Proses Login

Pada proses *login* aplikasi, diawali dengan memasukkan *input* berupa *username* dan *password* jika sudah memiliki akun. Jika login berhasil maka akan dilanjutkan ke halaman Utama, sedangkan jika *user* pembeli tidak memiliki akun, *user* dapat melakukan *register*.



Gambar 3.7. Flowchart Login

Di dalam proses login juga terdapat proses *register*, Pada proses *register* aplikasi, diawali dengan memasukkan *input* data seperti nama, nomor telpon, email, alamat, *username*, *password*, dan *confirm password*. Setelah memasukkan data yang diperlukan akan ada pengecekan terhadap data tersebut, seperti pengecekan email jika sudah di pakai dan pengecekan *username*. Jika tidak ada data yang sama, proses kan berlanjut ke halaman login.

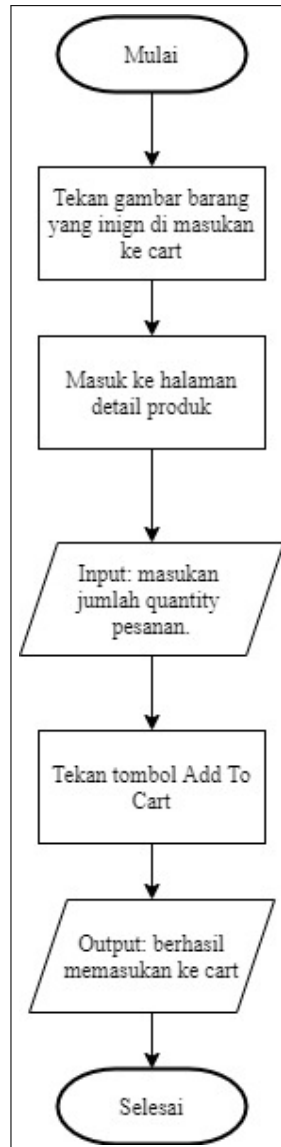


Gambar 3.8. Flowchart Register

## B Flowchart Tambah Barang Ke Cart

Dalam proses penambahan barang ke *cart* belanja, diawali dengan memilih produk yang dibeli, setelah memilih produk yang ingin dibeli *user* akan dialihkan

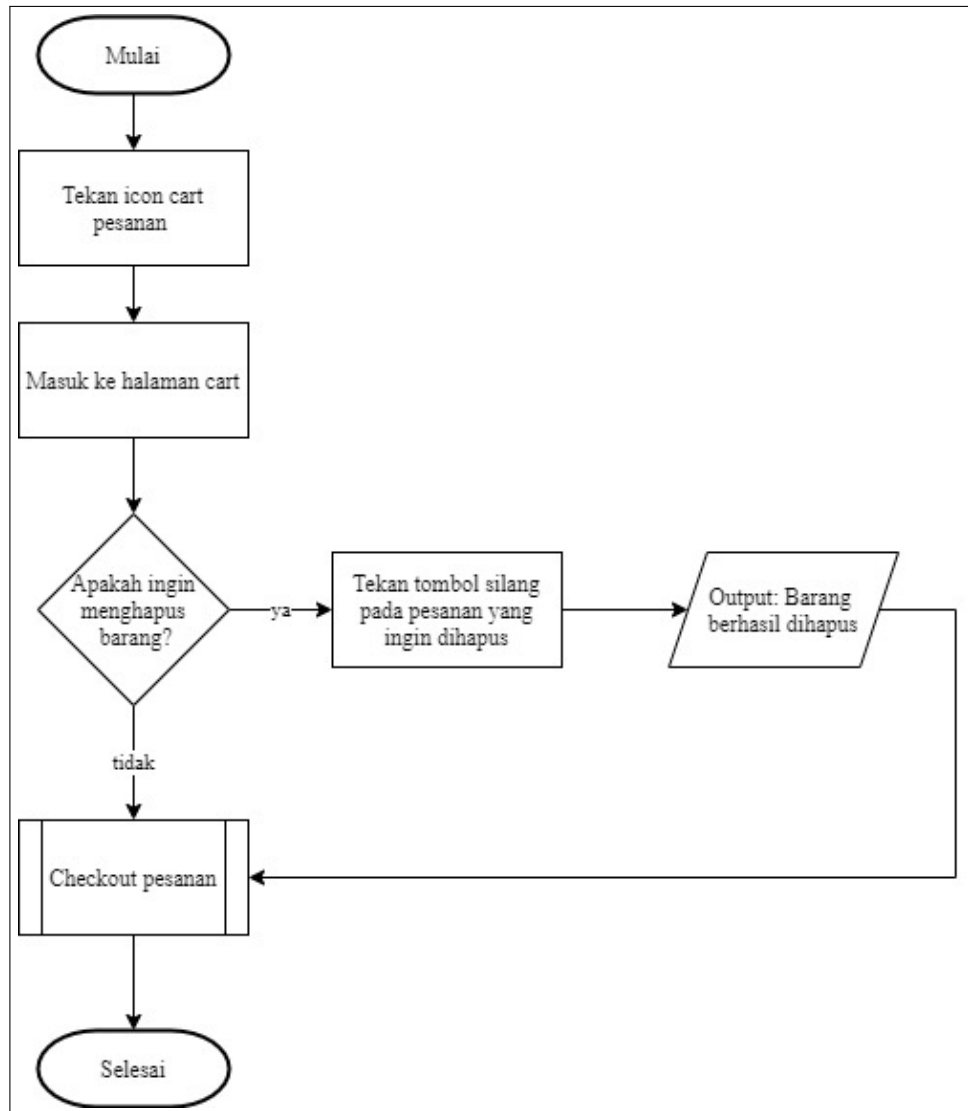
ke halaman detail produk. Lalu *user* juga akan diminta memasukkan jumlah barang yang ingin dibeli. Setelah memasukkan jumlah barang yang ingin dibeli *user* dapat menekan tombol *add to cart* dan pesanan berhasil tersimpan ke dalam *cart* belanja.



Gambar 3.9. Flowchart Tambah Barang Ke *Cart*

### C Flowchart *Cart* Barang Pesanan

Selanjutnya di dalam proses *cart* barang pesanan, pembeli dapat menghapus pesanan yang tidak jadi dibeli, jika pembeli sudah merasa puas dengan pilihannya, langkah selanjutnya adalah menekan tombol *checkout* pesanan.



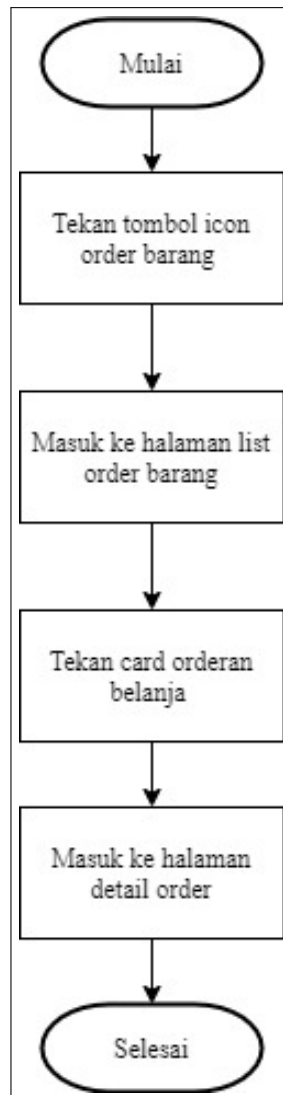
Gambar 3.10. Flowchart *Cart* Barang Pesanan

Setelah pembeli menekan tombol *checkout* pesanan, akan terjadi perpindahan halaman ke detail *checkout* pesanan. Di mana pembeli diminta untuk melakukan *input* berupa tanggal *pickup*, waktu *pickup*, dan *input* catatan.

#### D Flowchart Detail Order Barang

Dalam proses detail order barang pembeli dapat melihat list dari *history* pembelian yang sedang berlangsung dan pembelian yang sudah pernah dilakukan. Pembeli juga bisa melihat detail pembeliannya dengan cara menekan *cart* pesanan-

nya. Setelah menekan *cart* pesanan yang ingin dilihat, akan terjadi perpindahan halaman di mana halaman tersebut berisi detail tanggal transaksi tanggal pickup dan list barang yang dibeli.



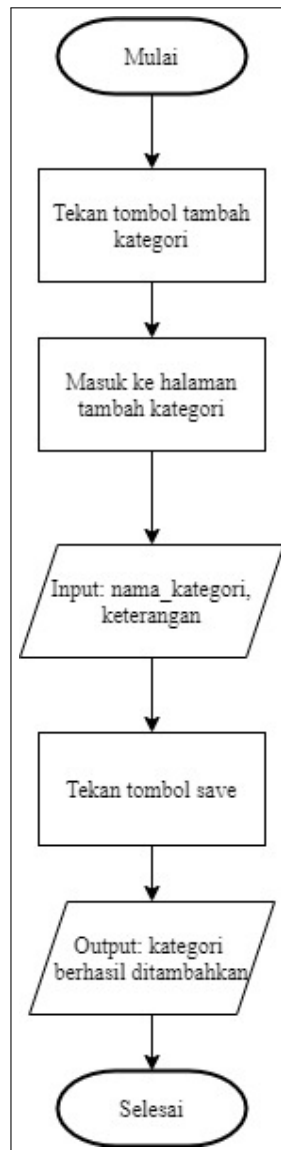
Gambar 3.11. Flowchart Detail Order Barang

## E Flowchart Tambah Kategori

Flowchart proses tambah kategori ini digunakan oleh admin di mana proses ini bertujuan untuk menambah macam - macam kategori yang akan dijual di dalam toko. Untuk penambahan kategori ini dibutuhkan *input* berupa nama kategori, dan keterangan. Hanya admin yang memiliki wewenang untuk melakukan penambahan



pada kategori, admin juga dapat menghapus kategori yang tidak diinginkan atau sudah tidak dijual lagi.

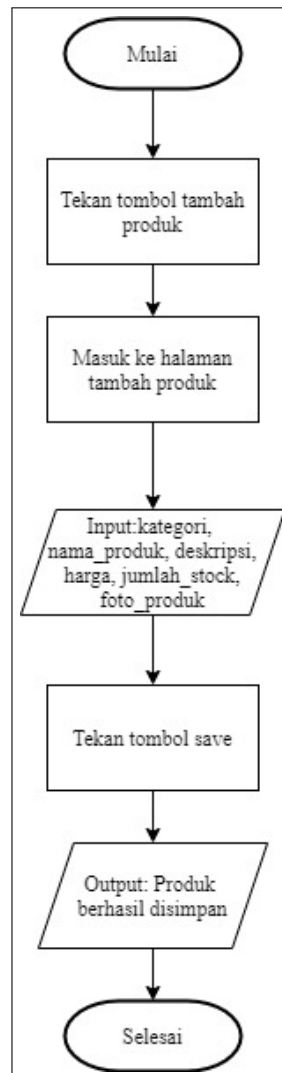


Gambar 3.12. Flowchart Tambah Kategori

## F Flowchart Tambah Produk

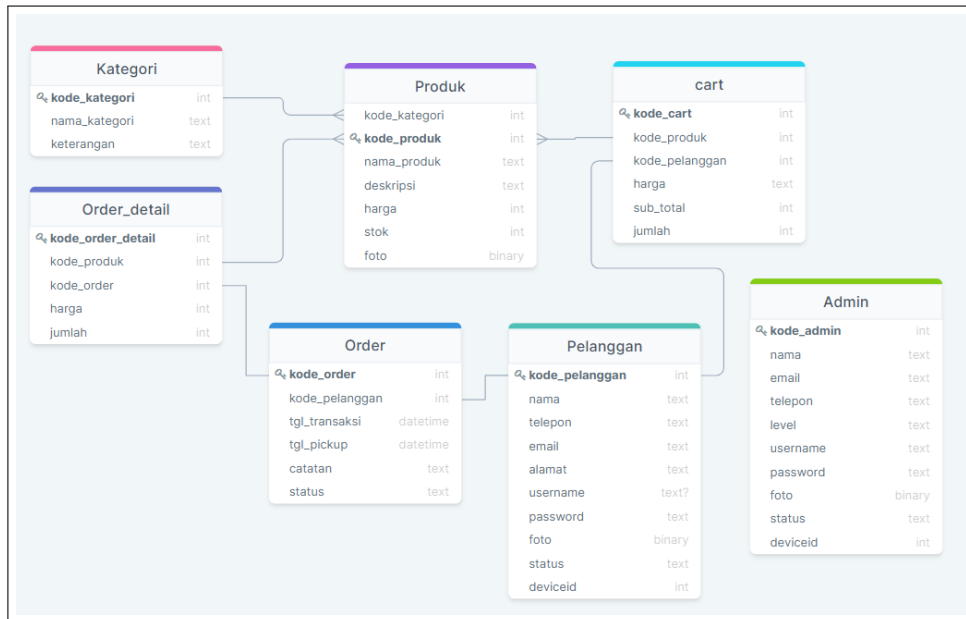
Flowchart proses tambah Produk ini juga hanya digunakan oleh admin di mana proses bertujuan untuk menambah berbagai macam produk sesuai kategori yang ada dan akan dijual di dalam toko. Admin juga butuh untuk melakukan *input* data berupa kategori, nama produk, deskripsi, harga, jumlah stok, dan foto produk

Admin juga dapat menghapus produk yang tidak diinginkan atau sudah tidak dijual lagi.



Gambar 3.13. Flowchart Tambah Produk

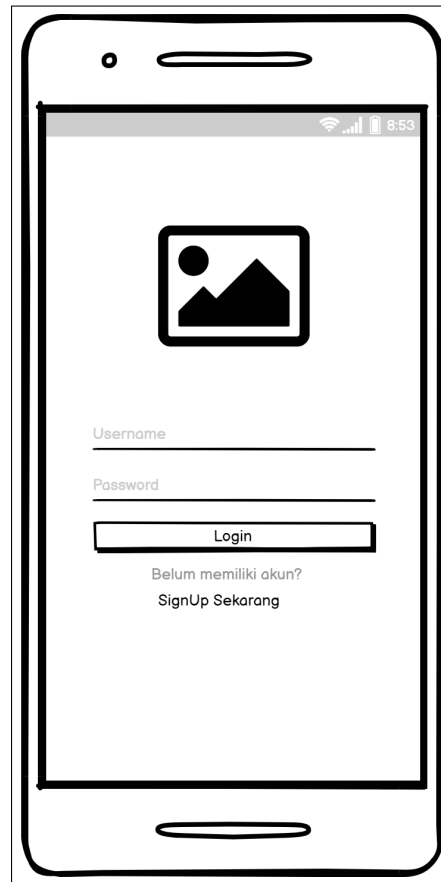
### 3.3.3 Database Schema



Gambar 3.14. Database Schema aplikasi

Pada Gambar 3.14 dapat dilihat bentuk dari *Database Schema* yang menampilkan entity yang digunakan pada saat pengerjaan aplikasi. Terdapat tabel admin, pelanggan, *cart*, order, order detail, kategori, dan produk. Di mana terdapat hubungan antara pelanggan dengan order dan *cart*, lalu ada hubungan antara produk dengan *cart*, kategori dan order detail. Dan hubungan antara order dengan order detail dan pelanggan.

### 3.3.4 Rancangan Antar Muka



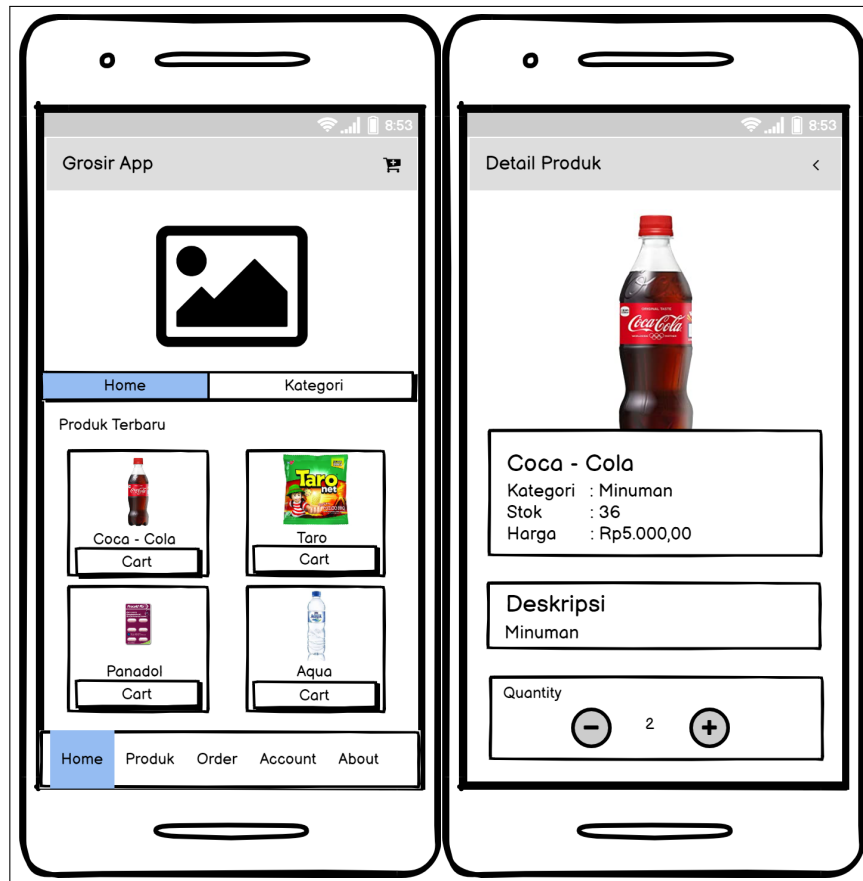
Gambar 3.15. Rancangan antar muka halaman Login

Pada Halaman login yang berisikan *input field* berupa *username* dan *password*. Jika sudah mengisi data yang diperlukan, klik *login* maka *user* akan dialihkan ke halaman dashboard berdasarkan *role* yang dimiliki, jika *user* belum memiliki akun yang terdaftar di aplikasi maka *user* dapat melakukan *SignUp* terlebih dahulu dengan menekan tulisan *SignUp* sekarang. Setelah *user* menekan tombol *SignUp* sekarang, *user* akan dialihkan ke halaman *register* akun.



Gambar 3.16. Rancangan antar muka halaman Register

Selanjutnya pada halaman *register* akun, untuk mendaftarkan akun baru *user* perlu memasukkan data diri yang diperlukan berupa nama, nomor telepon, email, *password*, alamat, *username*, *password*, dan *confirm password*. Setelah mengisi data yang diperlukan kemudian menekan tombol *register*, maka akan terjadi pengecekan email dan *username* apakah belum ada yang memakai atau sudah ada yang memakai, jika belum ada yang menggunakan *username* atau email yang dimasukan maka *user* akan berhasil mendaftarkan akun tersebut sebagai pembeli.



Gambar 3.17. Rancangan antar muka halaman *Home* dan Detail Barang (Pembeli)

Selanjutnya pada halaman *home* pembeli terdapat list tampilan Produk terbaru, di mana pembeli dapat melihat barang apa saja yang dijual di dalam toko tersebut, pembeli juga dapat langsung memasukkan barang yang diinginkan ke dalam *cart* dengan cara menekan tombol *cart* pada bagian bawah gambar, atau pembeli dapat menekan gambar barang yang diinginkan, lalu pembeli akan di alihkan ke halaman detail produk. Pembeli dapat melihat detail seperti Kategori, jumlah stok yang tersedia, harga barang dan deskripsi. Pembeli juga dapat memesan dengan cara memasukkan jumlah *quantity* dan menekan tombol *Add To Cart* yang berada di bawah *Quantity*.



Gambar 3.18. Rancangan antar muka halaman Kategori dan halaman Produk Perkategori (Pembeli)

Pada Halamanan kategori pembeli dapat melihat kategori apa saja yang dijual di toko tersebut. Untuk melihat barang apa saja yang ada di halaman kategori, pembeli dapat menekan tulisan kategori yang diinginkan. Lalu akan terjadi perpindahan halaman di mana halaman tersebut menampilkan barang berdasarkan kategori pilihan pembeli. Pada halaman ini pembeli juga dapat memesan barang sesuai keinginan.



Gambar 3.19. Rancangan antar muka halaman Produk (Pembeli)

Selanjutnya di halaman produk pembeli dapat melihat list dari seluruh barang yang dijual di dalam toko. Pembeli juga dapat memesan langsung barang yang diinginkan.





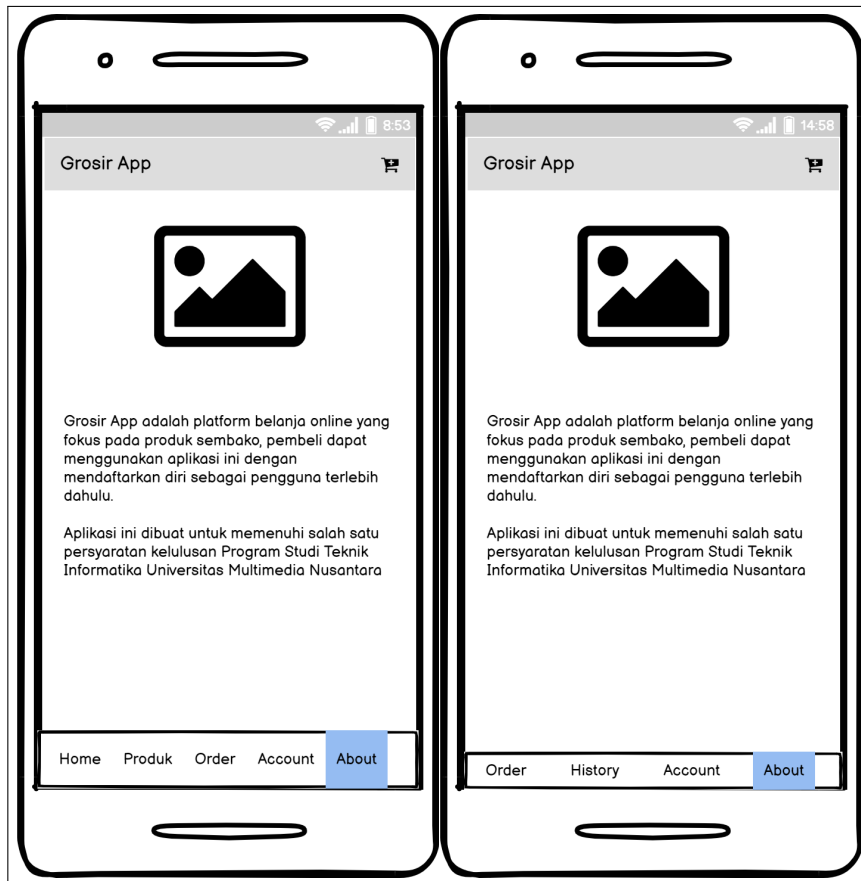
Gambar 3.20. Rancangan antar muka halaman Order dan Detail Order (Pembeli)

Untuk rancangan halaman Order dan Detail Order, pembeli dapat melihat list berupa transaksi orderan yang pernah dilakukan dan sudah selesai, lalu dapat melihat transaksi orderan yang baru saja dilakukan dan sedang dalam proses. Data yang dapat dilihat dalam list order berupa Kode orderan, Tanggal transaksi, Tanggal pickup, Total Belanja, dan status order. Sedangkan jika pembeli menekan salah satu data yang ada dalam list, maka akan dilanjutkan ke halaman Detail Order. Di dalam halaman detail order pembeli dapat melihat data sama seperti di list order, ditambah dengan data barang pesanan yang pernah dibeli, harga barang, dan *quantity* pesanan dan Total dari harga belanja.



Gambar 3.21. Rancangan antar muka halaman Account Pembeli dan Kasir

Selanjutnya terdapat dua halaman *Account* yaitu pada bagian *user* dan kasir, perbedaan halaman tersebut hanya pada bagian menu bar. Di halaman *Account* ini *user* dapat mengganti data nama, nomor telepon, email, dan *username*. Setelah melakukan *update* data, *user* dapat menekan tombol *update profile* dan data akan berubah. Terdapat juga tombol *logout* di mana jika *user* ingin keluar/mengganti akun *user* dapat menekan tombol *logout* yang nantinya akan kembali ke halaman login.



Gambar 3.22. Rancangan antar muka halaman About Pembeli dan Kasir

Pada Halaman *About* sama seperti halaman *Account* di mana isi halaman *About* Pembeli dan *About* kasir memiliki isi yang sama dan perbedaannya hanya ada pada bagian menu bar. Pada halaman *about* ini, halaman ini berisikan informasi mengenai aplikasi App Grosir yang dibuat.



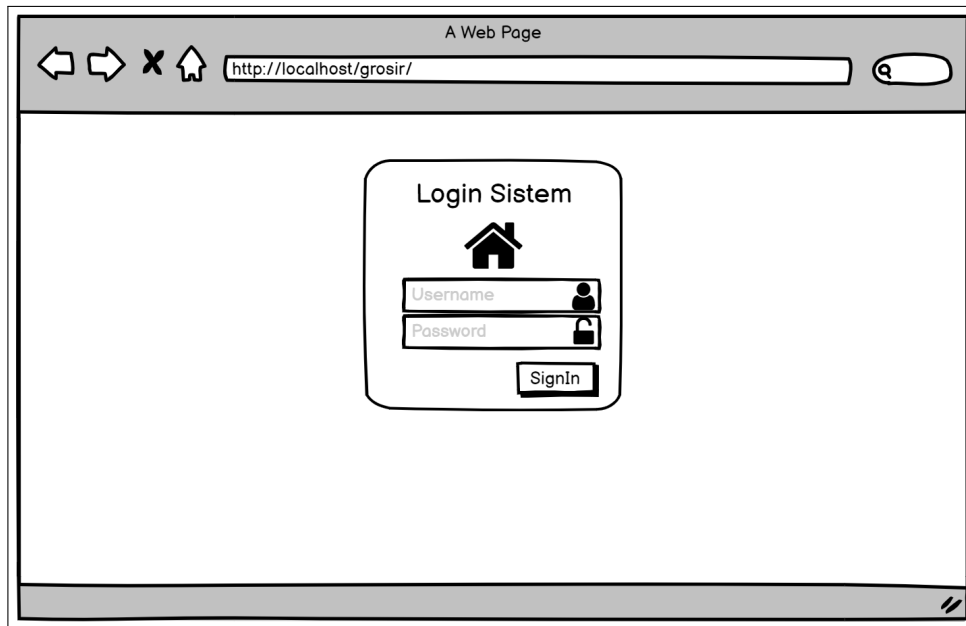
Gambar 3.23. Rancangan antar muka halaman Order dan Detail Order (Kasir)

Untuk halaman antar muka halaman order sebagai kasir, Halaman ini hanya berisikan list pesanan yang baru saja masuk yang harus segera di proses dan di siapkan barangnya oleh kasir untuk pembeli ambil. Kasir juga dapat menekan list orderan yang baru saja masuk, di mana *user* kasir akan dialihkan ke halaman detail order. Di dalam halaman detail order, *user* kasir dapat melihat data seperti kode order, tanggal transaksi, tanggal pickup, status order, nama pelanggan, data pesanan yang berisikan nama barang, harga barang, *quantity* barang dan total harga belanja.



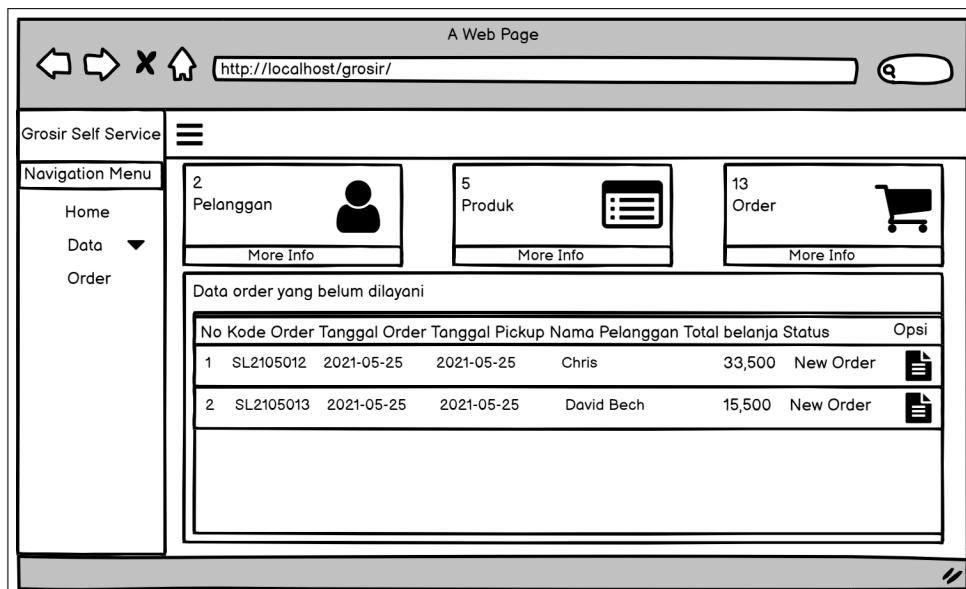
Gambar 3.24. Rancangan antar muka halaman History dan Detail Order History (Kasir)

Pada halaman history kasir, terdapat tampilan berupa list yang menampilkan data transaksi yang pernah dilakukan, di mana data transaksi itu berupa kode order, tanggal transaksi, tanggal pickup, total belanja, dan status order. Lalu jika kasir menekan salah satu data order yang pernah di lakukan akan membuka tampilan Detail order, di mana di halaman detail order kasir dapat melihat data sama seperti list dan ditambah nama pelanggan, nama barang, harga barang, *quantity* barang yang dipesan dan juga total harga dari belanjaan tersebut. Kasir juga dapat memberi tahu bahwa barang sudah siap untuk diambil dengan menekan tombol Ready yang nantinya pembeli akan mendapat notifikasi di aplikasinya bahwa pesanan sudah dapat diambil.



Gambar 3.25. Rancangan antar muka halaman Login (Admin)

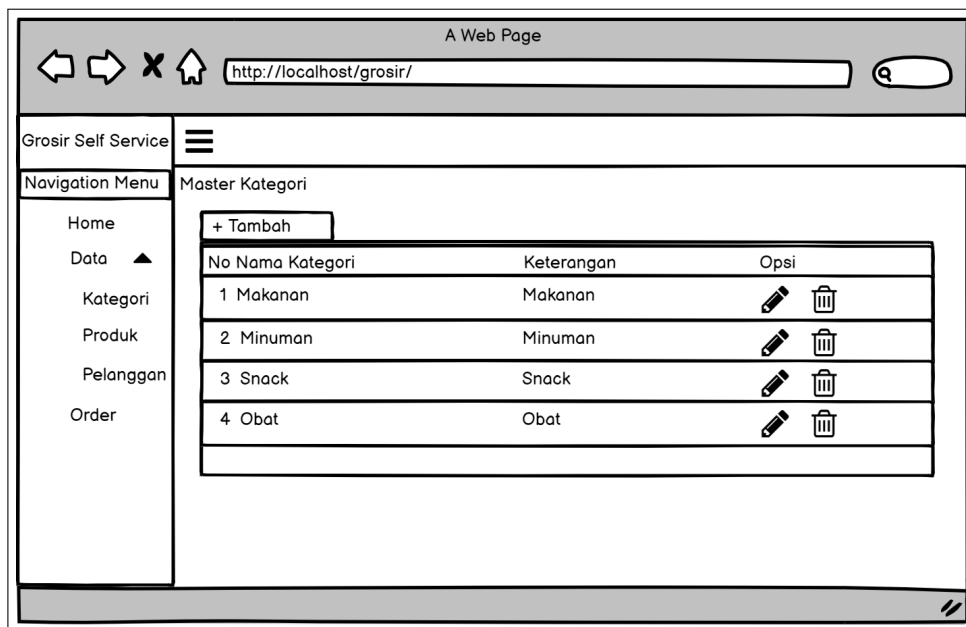
Untuk rancangan antarmuka halaman *login* admin berbasis web, admin diminta untuk melakukan *input* berupa *username* dan *password*. Setelah mengisi data berupa *username* dan *password* selanjutnya menekan tombol *SignIn* yang nantinya akan di alihkan ke halaman *home* admin



Gambar 3.26. Rancangan antar muka halaman *Home* (Admin)

Pada halaman *home* admin, admin dapat melihat total pelanggan yang sudah

melakukan *register* ke aplikasi, total produk yang dijualkan di toko, dan jumlah history order yang sudah pernah dilakukan. Di bawah itu juga terdapat list berupa data order masuk yang belum dilayani di mana data tersebut terdiri dari kode order, tanggal order, tanggal pickup, nama pelanggan, total belanja, dan status. terdapat juga opsi untuk melihat barang apa saja yang di pesanan dalam orderan tersebut, di mana untuk melihatnya dengan menekan tombol opsi yang ada disebelah list tersebut.



Gambar 3.27. Rancangan antar muka halaman Kategori (Admin)

Selanjutnya di halaman rancang antar muka pada halaman kategori admin, admin dapat melihat list kategori yang sudah pernah dibuat. Untuk membuka list tersebut, admin perlu menekan tombol data dan akan terjadi dropdown dan disana admin dapat menekan tombol kategori. Kategori yang sudah pernah dibuat dapat diubah dan dihapus dengan menggunakan tombol yang sudah disediakan. Admin juga dapat menambah kategori baru dengan cara menekan tombol tambah, yang nantinya akan membuka halaman Tambah kategori.

The screenshot shows a web browser window titled 'A Web Page' with the URL 'http://localhost/grosir/'. The application header is 'Grosir Self Service'. A navigation menu on the left lists: Home, Data (with a dropdown arrow), Kategori, Produk, Pelanggan, and Order. The main content area is titled 'Tambah Kategori'. It contains a 'Kembali' (Back) link with a left arrow, a text input field for 'Nama Kategori', a text area for 'Keterangan', and a 'Save' button.

Gambar 3.28. Rancangan antar muka halaman Tambah Kategori (Admin)

Selanjutnya setelah menekan tombol tambah kategori, admin akan masuk ke halaman Tambah kategori. Di mana halaman ini berisikan form untuk menambahkan kategori. Form tersebut meminta admin untuk mengisi data berupa nama kategori dan keterangan. Setelah itu admin dapat menekan tombol *save* untuk menyimpan kategori yang baru saja dibuat.

The screenshot shows the 'Data Produk' page. It features a '+ Tambah' button and a table with the following data:

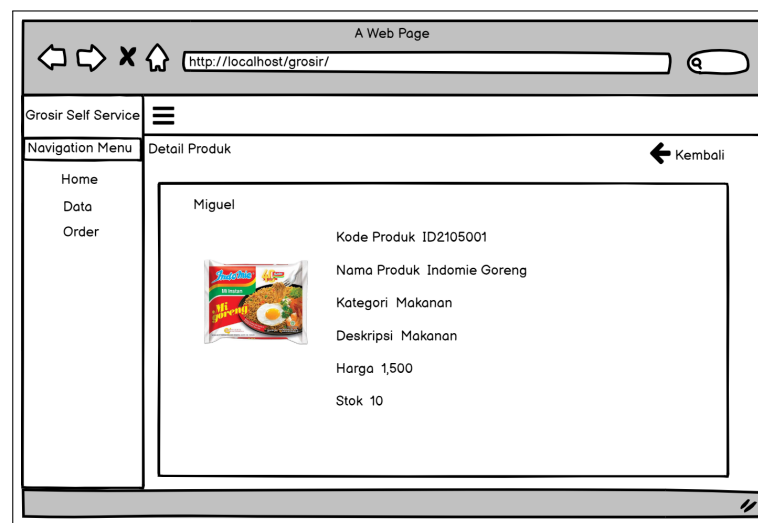
No	Foto	Kode Produk	Nama Produk	Kategori	Deskripsi	Harga	Stock	Terjual	Opsi
1		ID2105001	Indomie Goreng	Makanan	Makanan	1,500	25	15	
2		ID2105002	Coca - Cola	minuman	Makanan	4,500	50	29	
3		ID2105003	Lampu Philips	Listrik	Listrik	10000	10	2	

Gambar 3.29. Rancangan antar muka halaman Produk (Admin)

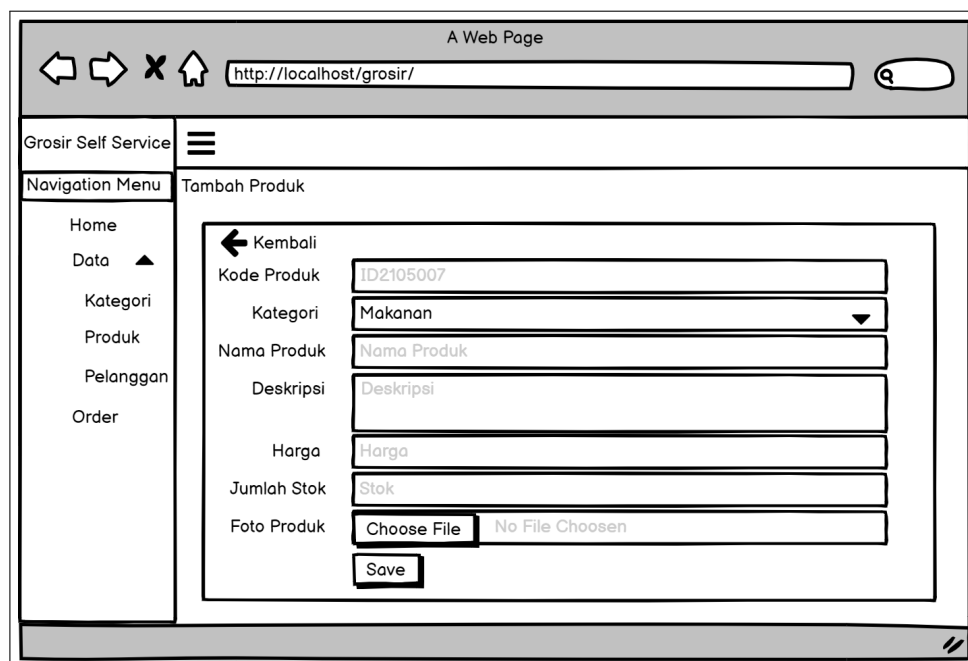
Setelah berhasil menambah kategori baru, admin dapat menambah produk



baru berdasarkan kategori yang sudah terdaftar, dan melihat list produk yang ada di toko, dengan menekan tombol produk. pada halaman produk ini admin dapat melihat produk yang ada di toko, di mana list tersebut menampilkan data berupa foto produk, kode produk, nama produk, kategori, deskripsi, harga, stok, jumlah terjualnya produk dan memiliki opsi untuk melakukan *edit* produk dan menghapus produk yang sudah tidak dijual, dan melihat detail produk seperti gambar dibawah.

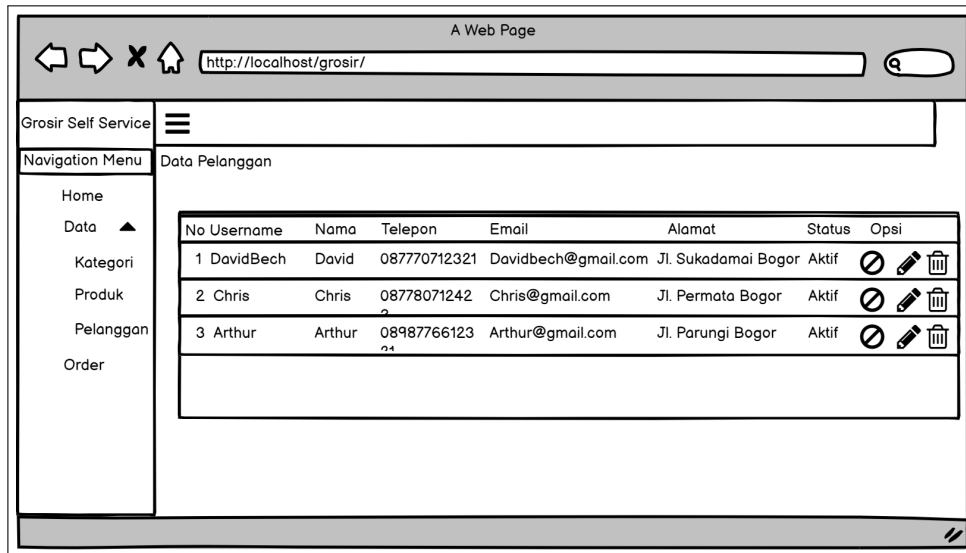


Gambar 3.30. Rancangan antar muka halaman Tambah Produk (Admin)



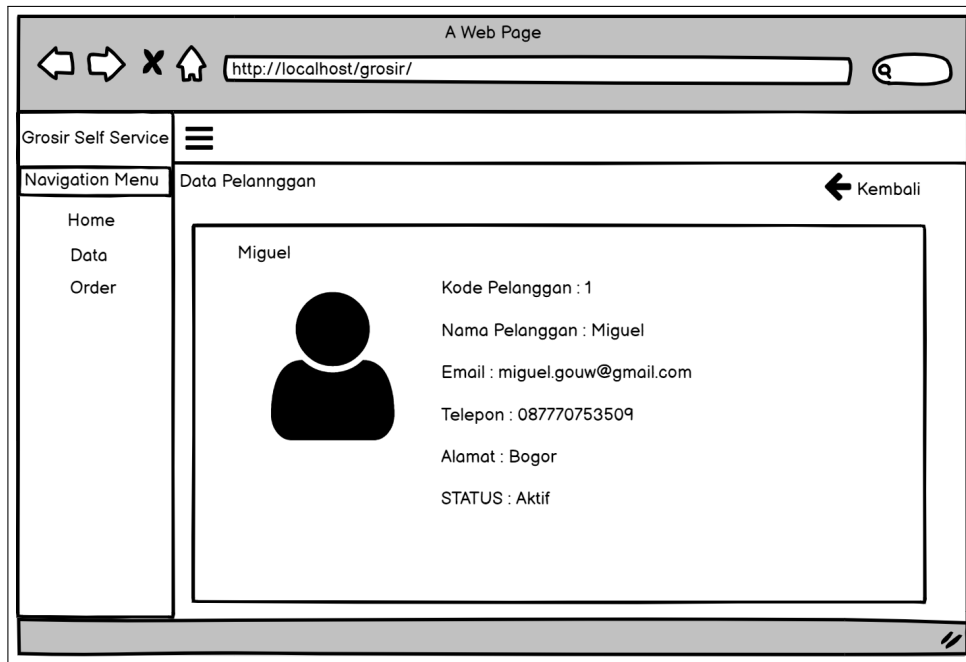
Gambar 3.31. Rancangan antar muka halaman Tambah Produk (Admin)

Pada Gambar 3.31 rancangan antar muka halaman tambah produk, admin dapat menambahkan data produk baru dengan melakukan *input* data kategori, nama produk, deskripsi, harga, jumlah stok dan foto produk. Setelah admin selesai mengisi data selanjutnya melakukan penyimpanan data dengan menekan tombol *save*.

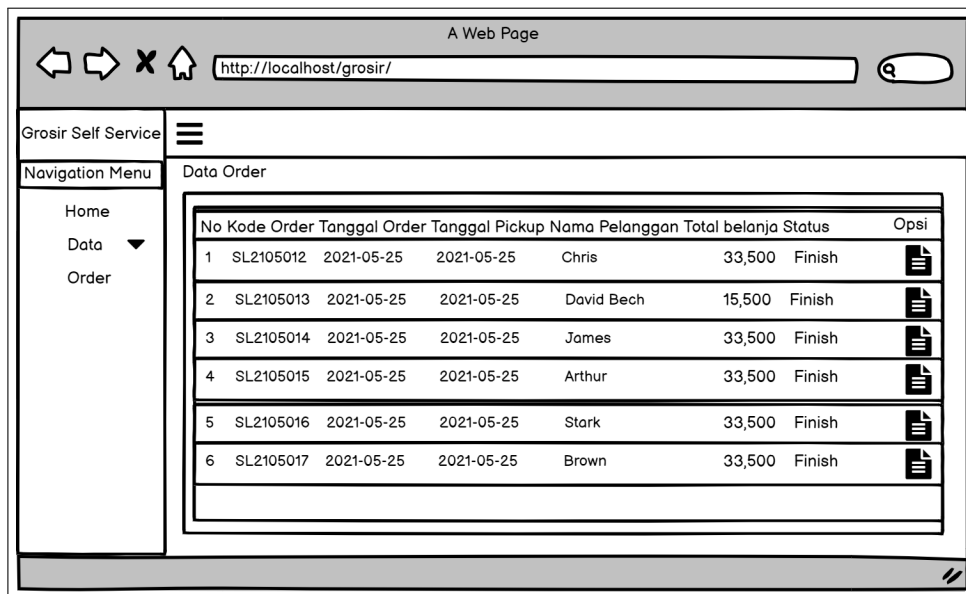


Gambar 3.32. Rancangan antar muka halaman Pelanggan Admin

Selanjutnya rancangan antar muka pada halaman pelanggan admin, disini admin dapat melihat data diri pelanggan yang sudah terdaftar, admin juga dapat menonaktifkan pelanggan, mengedit data pelanggan dan menghapus data pelanggan. Admin juga dapat melihat lebih detailnya dengan menekan salah satu data pelanggan yang nantinya admin akan di alihkan ke halaman detail pelanggan seperti gambar dibawah ini



Gambar 3.33. Rancangan antar muka halaman Detail Pelanggan Admin



Gambar 3.34. Rancangan antar muka halaman Order Admin

Pada Rancangan antar muka halaman order admin yang terlihat pada gambar diatas, disini admin dapat melihat list dari *history* dari pembelian yang dilakukan di mana di dalam list tersebut berisikan data berupa kode order tanggal order, tanggal pickup, nama pelanggan, total belanja dan status orderan. –